



Exploring High-Performance Architecture for Data Center Networks

Deshun Li¹, Shaorong Sun², Qisen Wu², Shuhua Weng¹, Yuyin Tan², Jiangyuan Yao^{1,*},
Xiangdang Huang¹ and Xingcan Cao³

¹School of Computer Science and Technology, Hainan University, Haikou, Hainan, 570228, China

²School of Cyberspace Security (School of Cryptology), Hainan University, Haikou, Hainan, 570228, China

³University of British Columbia, Vancouver, V6T1Z1, Canada

*Corresponding Author: Jiangyuan Yao. Email: yaojy@hainanu.edu.cn

Received: 15 July 2022; Accepted: 21 October 2022

Abstract: As a critical infrastructure of cloud computing, data center networks (DCNs) directly determine the service performance of data centers, which provide computing services for various applications such as big data processing and artificial intelligence. However, current architectures of data center networks suffer from a long routing path and a low fault tolerance between source and destination servers, which is hard to satisfy the requirements of high-performance data center networks. Based on dual-port servers and Clos network structure, this paper proposed a novel architecture *RClos* to construct high-performance data center networks. Logically, the proposed architecture is constructed by inserting a dual-port server into each pair of adjacent switches in the fabric of switches, where switches are connected in the form of a ring Clos structure. We describe the structural properties of *RClos* in terms of network scale, bisection bandwidth, and network diameter. *RClos* architecture inherits characteristics of its embedded Clos network, which can accommodate a large number of servers with a small average path length. The proposed architecture embraces a high fault tolerance, which adapts to the construction of various data center networks. For example, the average path length between servers is 3.44, and the standardized bisection bandwidth is 0.8 in *RClos*(32, 5). The result of numerical experiments shows that *RClos* enjoys a small average path length and a high network fault tolerance, which is essential in the construction of high-performance data center networks.

Keywords: Data center networks; dual-port server; clos structure; high-performance

1 Introduction

Data center networks (DCNs) accommodate a large number of servers and switches with high-speed links, which is an infrastructure of data centers to provide information services [1–3]. The structure of data center networks defines the connection relationship between switches and servers physically, which directly determines the level of service for users.



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

According to forwarding mechanisms, current data center networks can be classified into two categories: switch-centric and server-centric [4,5]. In switch-centric data center networks, the forwarding of data is completely borne by switches, where servers undertake data calculation and storage. The representative works include Fat-Tree [4], VL2 [6], Jellyfish [7], and S2 [8]. In server-centric data center networks, data forwarding is undertaken jointly by switch and server, or by multi-port server entirely where the switch is treated as a cross connector of the network. The representative works include DCell [5], BCube [9], FiConn [10] and CamCube [11,12]. However, the above server-centric architectures suffer from a large average path length and low fault tolerance, which is difficult to construct a high-performance data center network.

In this paper, we proposed a novel high-performance server-centric data center network *RClos*, which deploys commercial switches and dual-port servers to build its architecture. Logically, *RClos* is constructed by inserting a dual-port server into each pair of adjacent switches in the fabric of switches where switches are connected into a ring Clos structure. The proposed *RClos* inherits characteristics of the embedded ring Clos structure, which embraces a large scale of servers, a high bandwidth and a small network diameter. For example, *RClos*(24, 5) and *RClos*(48, 5) can accommodate 8640 and 69120 servers with a network diameter of 5, respectively. With a network diameter of 6, *RClos*(24, 7) and *RClos*(48, 7) can accommodate 145152 and 2322432 servers, respectively. We provided the connection and proved the properties of the *RClos* structure, which indicates the proposed architecture enjoys various advantages in constructing high-performance networks. We describe the structural properties of *RClos* in terms of network scale, bisection bandwidth, and network diameter in theory. The result of experiments on *RClos*(24, 5) shows that *RClos* enjoys a small average routing path, and it holds this property with a high server failure rate. The result demonstrates that *RClos* embraces a high transmission efficiency and fault tolerance, which can satisfy the requirements of high-performance data center networks.

This paper makes two contributions as follows. Firstly, we propose a high-performance structure *RClos* and present the detailed connection method through coordinates. Secondly, this paper describes the properties of *RClos* by theoretical proof and experimental results, which gives the structural performance of the proposed architecture objectively.

The rest of this paper is organized as follows. Section 2 presents the related work of data center networks, and Section 3 describes the construction of *RClos*. Section 4 proves the properties of *RClos*, and Section 5 presents simulation experiments. Section 6 summarizes this paper and describes future work.

2 Related Work

Based on the forwarding mechanism of networks, current architectures fall into switch-centric data center networks and server-centric data center networks. This section presents related work of data center networks in two categories, respectively.

2.1 Switch-Centric Data Center Networks

In switch-centric data center networks, servers undertake the function of data calculation and storage, and the task of data forwarding is entirely born by switches. A remarkable feature of this kind of networks is that each server is attached to a switch and does not participate in the interconnection between switches.

To solve the problems of scalability, network bandwidth, and single node failure, Fat-tree architecture was proposed, where the fabric of switches is a folded 5-stage Clos structure [4]. Based on Clos structure, servers in Fat-tree can achieve a bandwidth oversubscription ratio of 1:1. Therefore, Fat-tree can provide excellent network performance. In contrast, the structure and scale of this topology are limited by the number of ports in a switch and the average path length between servers is close to 6. Based on folded Clos, Monsoon adopts three-level switches between the aggregation and core level, and

adopts two-level switches between the access and aggregation levels [13]. To improve the scalability of the network and enhance the flexibility of dynamic resource allocation, VL2 (Virtual Layer 2) was proposed to support large-scale networks through a flexible expansion, which ensures high bandwidth between servers [6]. ElasticTree can achieve energy saving by adjusting its network topology dynamically [14]. To address the challenges of scale solidification in precise structures, Jellyfish was proposed to create a data center network with different scales and incremental deployment [7]. S2 is a flexible network constructed on top-of-rack switches, which can support coordinate-based greedy routing and multi-path routing with high scalability [8]. The latest work of switch-centric networks includes Hyper-network [15], HyScale [16], etc.

2.2 Server-Centric Data Center Networks

In server-centric data center networks, servers undertake both data calculation and forwarding. A remarkable feature is that each server is equipped with two or more network interface ports to participate in the network interconnection.

To solve the scalable challenges of traditional data center networks, DCell was proposed, which recursively defined its topology [5]. A high-level DCell is built from multiple lower-level DCells through complete interconnection. To satisfy the requirements of a modular data center network, BCube was proposed, where its topology is a generalized hypercube, and the adjacent servers are connected by an n -port switch [9]. FiConn was proposed [10] to avoid the overhead of installing more extra network interface cards (NICs). FiConn was built on the fact that each server is equipped with two NIC ports from the factory, one for connection and the other for backup. CamCube is a modular data center network that connects 6-port servers directly [11,12]. Each server in CamCube is equipped with six ports, and all servers are connected in a 3D Torus topology. DPillar [17] and HCN [18] are constructed on dual-port server and switch. SWCube [19] adopts hypercube topology on the dual-port servers, and SWDC [20] employs a small-work structure on the multi-port servers. The other work of switch-centric networks includes DPCell [21], HSDC [22], etc.

3 RClos Connection

In this section, we propose a server-centric data center architecture *RClos*. We first present the arrangement of devices in *RClos*, then describe the principle of the interconnection method.

3.1 Arrangement in *RClos*

RClos consists of two devices, the n -port switches and the dual-port servers. Servers and switches are arranged in m columns separately, where m is odd with $m \geq 3$. The number of n -port switches is $\left(\frac{n}{2}\right)^{\frac{m-1}{2}}$ in each switch column, and the number of dual-port servers is $\left(\frac{n}{2}\right)^{\frac{m+1}{2}}$ in each server column. Logically, the server column and switch column in *RClos* are arranged alternately in the form of a circular topology. Let $S_0 \sim S_{m-1}$ denote the server column $0 \sim m-1$, and $W_0 \sim W_{m-1}$ denote the switch column $0 \sim m-1$, respectively. Fig. 1 presents a vertical view of the switch column and the server column arrangement. We can see from the vertical view, server column S_i is adjacent to switch column W_i and $W_{(i+1) \bmod m}$ on left and right sides, respectively; and switch column W_i is adjacent to server column $S_{(i-1) \bmod m}$ and S_i on the left and the right sides, respectively.

Fig. 2 provides a plan view of a *RClos* network with $m = 5$ and $n = 4$. From Fig. 2, each switch is identified by its row and column coordinate. Let (w_1, w_2) denote a switch located in row w_1 and column w_2 , where w_1 and w_2 take values from $\left[0, \left(\frac{n}{2}\right)^{\frac{m-1}{2}}\right)$ and $[0, m)$, respectively. Similarly, each server is

identified by its row and column coordinates. Let (s_1, s_2) denote the server located in row s_1 and column s_2 , where s_1 and s_2 take values from $\left[0, \binom{n}{2} \frac{m+1}{2}\right)$ and $[0, m)$, respectively.

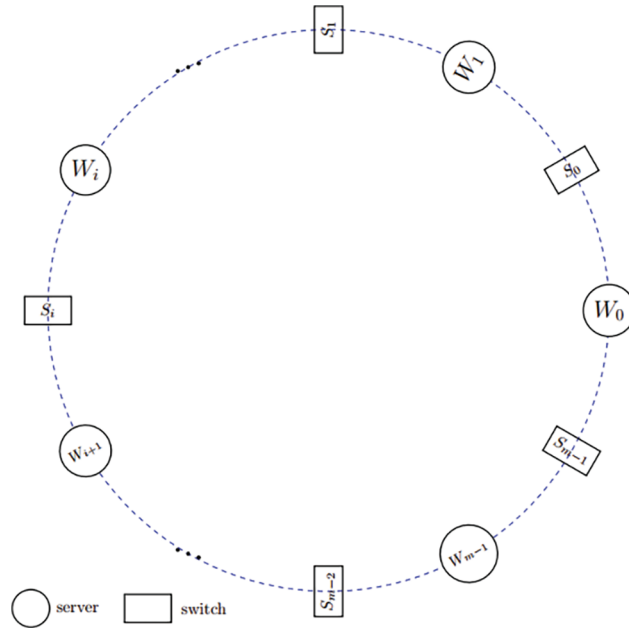


Figure 1: The vertical view of the *RClos* network structure

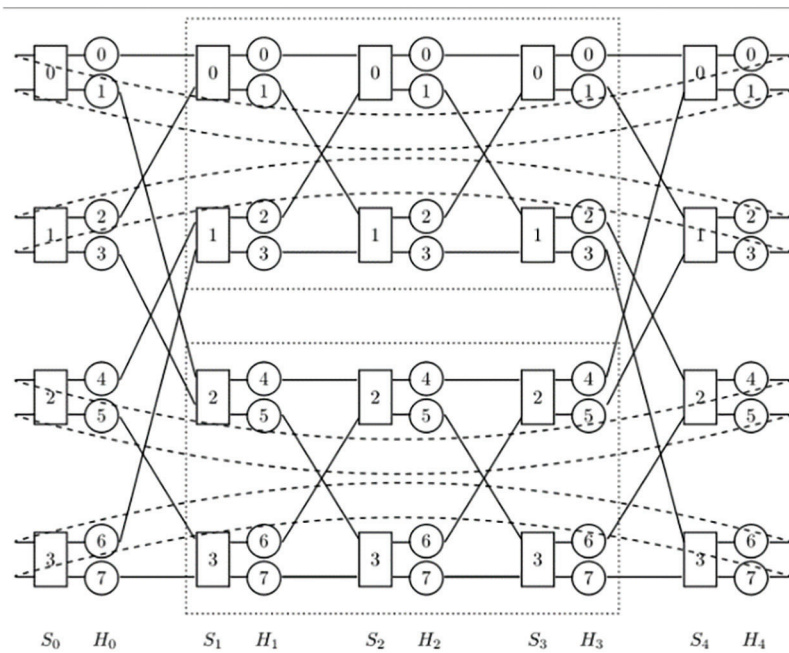


Figure 2: The plan view of a *RClos* network structure

From Fig. 2 and the above description, we learn that each *RClos* network is defined by n and m , where n is the number of ports in a switch and m is the number of columns of switch or server in *RClos*. Thus, let $RClos(n, m)$ represent the *RClos* network composed of m -column servers and switches. Fig. 2 provides a plan view of $RClos(4, 5)$, which includes five columns of switches and five columns of servers. Without consideration of these dual-port servers, the fabric of switches in $RClos(4, 5)$ forms a circular interconnected 5-level Clos structure.

3.2 *RClos* Connection

For the convenience of description, we divide ports in a switch into the left and the right side as shown in the plan view, and each part contains a half the number of ports, i.e., $\frac{n}{2}$. Switch ports in each part are numbered from top to bottom, which takes value from $\left[0, \frac{n}{2}\right)$. For server (s_1, s_2) and switch (w_1, w_2) , we describe the interconnection based on coordinates.

Rule 1: a server connects to the switch in its left switch column. In server column S_i , each server (s_1, s_2) connects to the right port of its left switch column. Server (s_1, s_2) connects to switch (w_1, w_2) with $w_1 = s_1$, and $w_2 = \frac{s_2 - \left(s_2 \bmod \frac{n}{2}\right)}{n/2}$.

The connection rules of the switch on the right side of a server is divided into three parts:

Rule 2: a server in column S_0 connects to the switch in column H_1 . A server $(s_1, 0)$ in column S_0 connects to switch (w_1, w_2) in column H_0 , with $w_1 = \left\lceil s_1 - \left(s_1 \bmod \frac{n^2}{4}\right) \right\rceil / \frac{n^2}{4} + \left(\frac{n}{2}\right)^{\frac{m-3}{2}} \cdot \left(s_1 \bmod \frac{n}{2}\right)$ and $w_2 = 1$.

Rule 3: server in column S_i connects to switch in column W_{i+1} with $i \in [1, m - 1)$. As shown in Fig. 3, without the connection between column S_{m-1} and column W_{m-1} , $RClos(n, m)$ can be treated as a modular recursive interconnection of $C(n, m)$, which contains server columns S_0 and S_{m-1} , switch columns H_0 and H_{m-1} , and modular $C(n, m - 2)$ from 0 to $\frac{n}{2} - 1$. In each $C(n, m - 2)$, servers in the first column use Rule 2 to connect to its left switch column. This process recursively continues until server column $S_{\frac{n-3}{2}}$. For server column S_i with $i \in \left[\frac{m-1}{2}, m - 1\right)$ and switch column W_{i+1} , the connection is fully symmetric with that from server column S_{m-s_2-1} to switch column W_{m-s_2} .

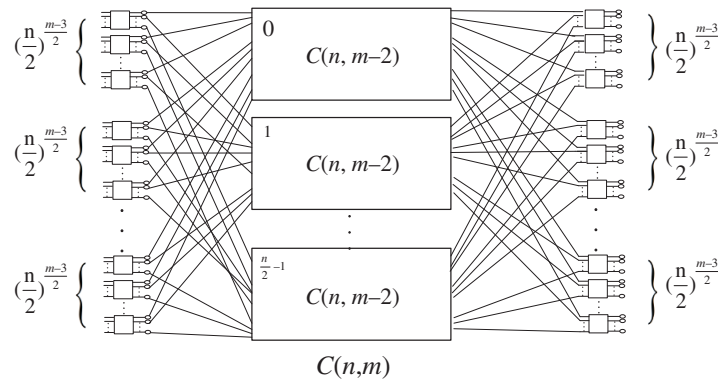


Figure 3: The modular recursive interconnection of $C(n, m)$

Rule 4: server in column S_{m-1} connects to switch in column W_0 . Server $(s_1, m-1)$ in column S_{m-1} directly connects to switch $(w_1, 0)$ in column W_0 with $w_1 = \frac{s_1 - \left(s_1 \bmod \frac{n}{2}\right)}{n/2}$.

Through the above connection, the fabric of switches forms a ring Clos network structure. Logically, a *RClos* structure is built by inserting a dual-port server into two adjacent switches in the ring Clos network. Based on the dual-port server, *RClos* inherits advantages and characteristics of the Clos structure, which can achieve a high performance in the resulting architecture of the data center network.

4 Properties of RClos

In this section, we describe the structural properties of *RClos* in terms of network scale, bisection bandwidth, and network diameter.

4.1 Network Scale

Network scale is a key concern of data center network performance. The number of servers reflects the computing and storage capability of a data center, which represents the scalability of the network structure.

Theorem 1: In a $RClos(n, m)$ network, the number of n -port switches is $N_{n,m} = m \left(\frac{n}{2}\right)^{\frac{m-1}{2}}$, and the number of dual-port servers is $T_{n,m} = m \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$.

Proof: According to its arrangement, the number of server columns and switch column is m in $RClos(n, m)$. From the construction, we learn that the number of switches in each column is $\left(\frac{n}{2}\right)^{\frac{m-1}{2}}$, and the number of servers in each column is $\left(\frac{n}{2}\right)^{\frac{m+1}{2}}$. Therefore, the total number of switches is $N_{n,m} = m \left(\frac{n}{2}\right)^{\frac{m-1}{2}}$ and the total number of servers is $T_{n,m} = m \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$. End Proof.

As we learn from Theorem 1, with given m , the number of servers $T_{n,m}$ increases in power law with the number of switch ports n . With a given n , the number of servers $T_{n,m}$ increases in a compound law of linearity and exponentially with the number of columns m . This indicates that *RClos* enjoys excellent scalability to construct large-scale data center networks. For example, given $n = 24$, $RClos(24, 5)$ built on $m = 5$ can accommodate 720 24-port switches and 8640 dual-port servers; with $n = 48$, the number of switches and servers in $RClos(48, 5)$ is 2880 and 69120, respectively. Given $m = 7$, $RClos(24, 7)$ can accommodate 12096 24-port switches and 145152 dual-port servers; it is 96768 and 2322432 in $RClos(48, 5)$, respectively. With two alterable parameters n and m , $RClos(n, m)$ can satisfy the demand of various data center networks scale.

4.2 Bisection Bandwidth

Bisection bandwidth is the minimum number of links to be cut off to divide the network into two equal parts. A large bisection bandwidth indicates that the network architecture embraces excellent fault tolerance. For bisection bandwidth in $RClos(n, m)$, we have the following theorem:

Theorem 2: The bisection bandwidth of $RClos(n, m)$ is $B_{n,m} = 2 \cdot \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$.

As we learn from the construction, $RClos(n, m)$ is symmetrical. Therefore, we can adopt the same method in [17,23] to split a network into two parts in both vertical and horizontal directions to obtain bisection bandwidth.

Proof: Considering the completely non-block properties of Clos structure, we learn that there is a loop between each pair of switches in column W_0 and W_{m-1} through a switch in the middle column $W_{\frac{m-1}{2}}$. In a $RClos(n, m)$, there are at least $\left(\frac{n}{2}\right)^{\frac{m+1}{2}}$ circle paths of this kind. Therefore, to divide a $RClos(n, m)$ into two equal parts in a vertical direction, each circle path will be cut off at least two times. Thus, we get the number $2 \cdot \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$ in vertical bisection.

Considering the bisection of a $RClos(n, m)$ in a horizontal bisection. When $n = 4k$, we know that $\left(\frac{n}{2}\right)^{\frac{m+1}{2}}$ is even. Thus, it needs to cut off $2 \cdot \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$ links to divide $RClos(n, m)$ into two equal parts, where the links locate between server column S_0 and switch column W_1 , and between server column S_{m-2} and switch column W_{m-1} . When $n = 4k + 2$, $\left(\frac{n}{2}\right)^{\frac{m+1}{2}}$ is odd. To divide a $RClos(n, m)$ into two equal parts, one should cut off $2 \cdot \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$ as that in $n = 4k$ and divide a module $C(n, m - 2)$ recursively in horizontal bisection. Thus, the total quantity of links to be cut off is greater than $2 \cdot \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$. Consider the result in horizontal and vertical bisection, we get $B_{n,m} = 2 \cdot \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$ in $RClos(n, m)$. End Proof.

With the number of servers $m \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$, Theorem 2 shows that the bisection bandwidth of $RClos(n, m)$ is $2 \cdot \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$. Thus, the standardized bisection bandwidth is $\frac{4}{m}$. For example, the bisection bandwidth of $RClos(24, 5)$ is 3456, with a standardized bisection bandwidth of 0.8. Considering the growth law of the number of servers, the $RClos$ enjoys a large bisection bandwidth, which indicates that the $RClos$ can provide fault tolerance and multi-path routing.

4.3 Path Length

Network diameter is the maximum value of the shortest path length between any two servers, where a small diameter can shorten transmission delay and improve transmission efficiency. For the maximum path length between any two switches in a $RClos(n, m)$, we have the following theorem:

Theorem 3: In a $RClos(n, m)$, the maximum path length between two switches is $d_{n,m} = m - 1$.

Proof: From the plan view of $RClos(n, m)$, we learn that the network is symmetric about the switch column $\frac{m-1}{2}$. The path length from switch column $\frac{m-1}{2}$ to the symmetric column on both sides is equal. Consider the fabric of switches without servers, let $d_{n,m}$ denote the maximum path length between two switches in a $C(n, m)$ module. According to the construction process, we have $d_{n,m} = 2 + d_{n,m-2}$. According to the Clos structure, we have $d_{n,3} = 2$. Therefore, we have $d_{n,m} = m - 1$ in a $RClos(n, m)$. End Proof.

From the perspective of the server, the longest path between any two dual-port servers will pass through at most $d_{n,m} = m - 1$ switches in a $RClos(n, m)$. Therefore, we have the following corollary:

Corollary 1: In a $RClos(n, m)$, the network diameter $D_{n,m} = m$.

The above theorem and corollary show that the network diameter of $RClos(n, m)$ increases linearly with the number of columns m . Consider network scale $T_{n,m} = m \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$, large-scale $RClos(n, m)$ can maintain a small network diameter with an increment of the number of servers. For example, $RClos(48, 7)$ can

accommodate millions of servers with a diameter of 7; $RClos(n, 5)$ with a diameter of 5 can meet the demand of tens of thousands of servers. $RClos(n, m)$ achieves a large network scale with a small diameter, which is essential in the construction of large-scale and high-performance data center networks.

4.4 Network Scale with Given Diameter

With a given network diameter, we consider the number of servers that a $RClos(n, m)$ can accommodate. Given network diameter d , we have the following theorem:

Theorem 4: Given network diameter d , a $RClos$ can accommodate $d \left(\frac{n}{2}\right)^{\frac{d+1}{2}}$ servers, where d is an odd number.

Proof: According to Theorem 1 and Corollary 1, we substitute network diameter $d = m$ into the formula $T_{n,m} = m \left(\frac{n}{2}\right)^{\frac{m+1}{2}}$, then we obtain the number $d \left(\frac{n}{2}\right)^{\frac{d+1}{2}}$. End Proof.

This theorem shows that given network diameter d , the number of servers increases on power law with the number of ports n in a switch in $RClos$. Given $d = 5$, the number of servers in $RClos$ is 8640 when $n = 24$, and it is 69120 when $n = 48$. Given a network diameter $d = 7$, the number of servers is 145152 and 2322432 when $n = 24$ and $n = 48$, respectively. Considering the growth law of network scale, $RClos(n, 5)$ or $RClos(n, 7)$ can satisfy the construction of large-scale data centers.

5 RClos Experiments

In this section, we conduct numerical experiments on the proposed $RClos$, including the path length under different scales of network, and the path length under different ratios of server failure.

5.1 Path Length

To verify the advantage of the short path in $RClos(n, m)$, we study the path length between switches and servers, and the path length distribution between servers in $RClos(n, 5)$.

Fig. 4 presents the average path length between servers and switches in $s(n, 5)$. As we can see, the average path length between servers and switches increases with the number of ports n in a switch, with a diameter of 5 in $RClos(n, 5)$. With the number of ports n in a switch increasing from $n = 8$ to $n = 32$, the average path length between servers increases by 10.6% to 3.44, and the average path length between switches increases by 9.7% to 3.10. In comparison, Fat-tree has an average path length close to 6 with a network diameter of 6. Considering that a $RClos(n, 5)$ can accommodate hundreds of thousands of servers, the path length is small in $RClos(n, 5)$, which contributes to higher performance routing transmission.

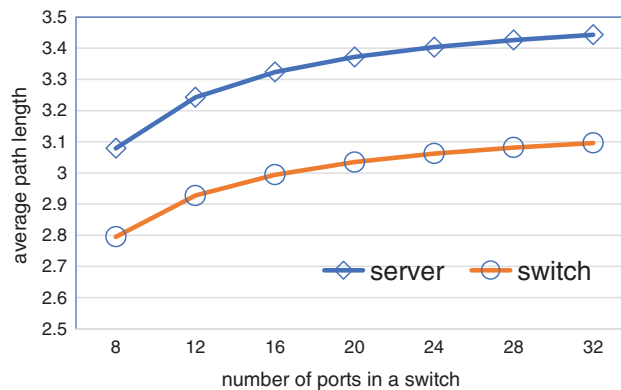


Figure 4: The average path length between servers and switches in $RClos(n, 5)$

Fig. 5 provides the histogram of path length distribution between servers and switches in $RClos(16, 5)$ and $RClos(24, 5)$. Fig. 5 shows that the distance between servers in $RClos(n, 5)$ is concentrated on 3 and 4, and only a small path length ratio reaches the network diameter of 5. The path length distribution between switches is similar, while the maximum path length between switches is 4. By studying path length distribution between servers in $RClos(16, 5)$ and $RClos(24, 5)$, we can find that the proportion of short path decreases with the increment of the number of ports n in a switch, and the proportion of long path increases. For example, in $RClos(16, 5)$ and $RClos(24, 5)$, the proportion of path of length 2 is about 8.12% and 11.8%, while the proportion of path with a length of 4 is about 39.4% and 42.2%, respectively. With an increment of the number of ports in a switch, a large proportion of long paths leads to an increment in average path length between servers.

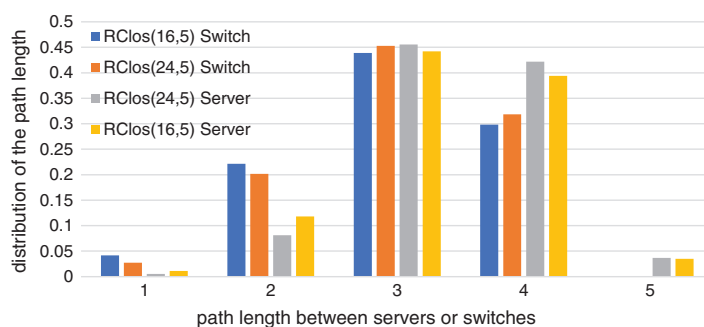


Figure 5: The path length distribution between switches and servers in $RClos(16, 5)$ and $RClos(24, 5)$

5.2 Fault Tolerance

Fig. 6 presents the network diameter and path length with the ratio of server failure in $RClos(16, 5)$ and $RClos(24, 5)$. With a given failure ratio, the result is an average of 10 rounds. With an increment of the ratio of server failure, the average path length between servers increases slightly, while the network diameter increases significantly. As the ratio of server failure increases from 0% to 16%, the path length between servers increases by 32.0% to 4.49%. The path length between servers in $RClos(24, 5)$ increases by 42.1% to 4.73%. Thus, the proposed network embraces an excellent fault tolerance, which maintains a small average path length between servers under a large proportion of server failure.

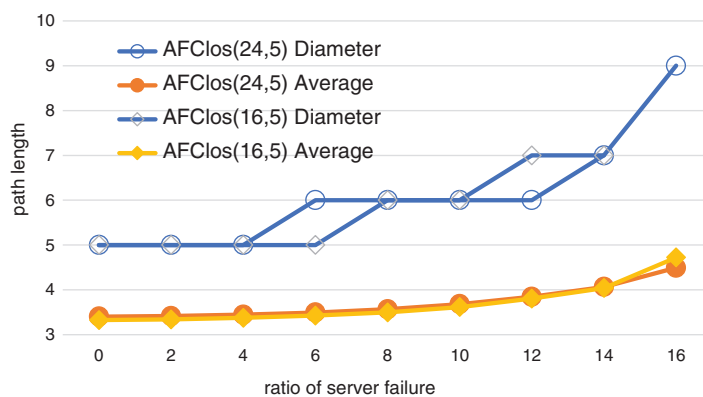


Figure 6: The network diameter and path length with the ratio of server failure in $RClos(16, 5)$ and $RClos(24, 5)$

The experimental results show that $RClos(n, m)$ can provide a small network diameter and average path length between servers, and also maintains the property under a high ratio of server failure. Therefore, $RClos(n, m)$ can construct data center networks with a high network performance and a high fault tolerance.

6 Conclusion

To settle the challenges of the long routing paths and low fault tolerance, this paper proposed $RClos$ to meet the construction requirements of a high-performance data center network. Logically, the fabric of switches in $RClos$ forms a ring Clos network, and a $RClos$ is built by inserting a dual-port server into adjacent switches. $RClos$ inherits the characteristics of its embedded Clos structure, which embraces a large network scale, a short routing path, and a high fault tolerance. For example, $RClos(24, 5)$ and $RClos(48, 5)$ can accommodate 8640 and 69120 servers with a network diameter of 5, respectively. The average path length between servers is 3.44, and the standardized bisection bandwidth is 0.8 in $RClos(32, 5)$. In future work, we will focus on routing scheduling mechanisms and multi-path routing in application deployment [24,25].

Funding Statement: This work was supported by the Hainan Provincial Natural Science Foundation of China (620RC560, 2019RC096, 620RC562), the Scientific Research Setup Fund of Hainan University (KYQD(ZR)1877), the National Natural Science Foundation of China (62162021, 82160345, 61802092), the key research and development program of Hainan province (ZDYF2020199, ZDYF2021GXJS017), and the key science and technology plan project of Haikou (2011-016).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] R. Jayamala and A. Valarmathi, "An enhanced decentralized virtual machine migration approach for energy-aware cloud data centers," *Intelligent Automation & Soft Computing*, vol. 27, no. 2, pp. 347–358, 2021.
- [2] Z. Wang, H. Zhang, X. Shi, X. Yin, H. Geng *et al.*, "Efficient scheduling of weighted coflows in data centers," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 9, pp. 2003–2017, 2019.
- [3] Y. Sanjalawe, M. Anbar, S. Al-Emari, R. Abdullah, I. Hasbullah *et al.*, "Cloud data center selection using a modified differential evolution," *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3179–3204, 2021.
- [4] M. Al-Fares, A. Loukissas and A. Vahdat, "A scalable, commodity data center network architecture," in *Proc. of ACM SIGCOMM*, Seattle, WA, USA, pp. 63–74, 2008.
- [5] C. Guo, H. Wu and K. Tan, "DCell: A scalable and fault-tolerant network structure for data centers," in *Proc. of ACM SIGCOMM*, Seattle, WA, USA, pp. 75–86, 2008.
- [6] A. Greenberg, J. R. Hamilton and N. Jain, "VL2: A scalable and flexible data center network," in *Proc. of ACM SIGCOMM*, Barcelona, Spain, pp. 51–62, 2009.
- [7] A. Singla, C. Y. Hong and L. Popa, "Jellyfish: Networking data centers randomly," in *Proc. of USENIX NSDI*, Seattle, WA, USA, pp. 225–238, 2012.
- [8] Y. Yu and C. Qian, "Space shuffle: A scalable, flexible, and high-bandwidth data center network," in *Proc. of IEEE ICNP*, Raleigh, NC, USA, pp. 13–24, 2014.
- [9] C. Guo, G. Lu and D. Li, "BCube: A high performance, server-centric network architecture for modular data centers," in *Proc. of ACM SIGCOMM*, Beijing, China, pp. 63–74, 2019.
- [10] D. Li, C. Guo and H. Wu, "FiConn: Using backup port for server interconnection in data centers," in *Proc. of IEEE INFOCOM*, Rio de Janeiro, Brazil, pp. 2276–2285, 2009.
- [11] H. Abu-Libdeh, P. Costa and A. Rowstron, "Symbiotic routing in future data centers," in *Proc. of ACM SIGCOMM*, Delhi, India, pp. 51–62, 2010.

- [12] P. Costa, A. Donnelly and G. O'Shea, "CamCube: A key-based data center," in *Technical Report MSR TR-2010-74*, Microsoft Research, 2010.
- [13] A. Greenberg, P. Lahiri and D. A. Maltz, "Towards a next generation data center architecture: Scalability and commoditization," in *Proc. of PRESTO*, Seattle, Washington, USA, pp. 57–62, 2008.
- [14] B. Heller, S. Seetharaman and P. Mahadevan, "ElasticTree: Saving energy in data center networks," in *Proc. of USENIX NSDI*, California, USA, pp. 249–264, 2010.
- [15] G. Qu, Z. Fang, J. Zhang and S. -Q. Zheng, "Switchcentric data center network structures based on hyper-graphs and combinatorial block designs," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 4, pp. 1154–1164, 2015.
- [16] S. Saha, J. S. Deogun and L. Xu, "Hyscale: A hybrid optical network based scalable, switch-centric architecture for data centers," in *Proc. of ICC*, Ottawa, Canada, pp. 2934–2938, 2012.
- [17] Y. Liao, J. Yin and D. Yin, "DPillar: Dual-port server interconnection network for large scale data centers," *Computer Networks*, vol. 56, no. 8, pp. 2132–2147, 2012.
- [18] D. Guo, T. Chen and D. Li, "Expandable and cost-effective network structures for data centers using dual-port servers," *IEEE Transactions on Computers (TC)*, vol. 62, no. 7, pp. 1303–1317, 2014.
- [19] D. Li and J. Wu, "On the design and analysis of data center network architectures for interconnecting dual-port servers," in *Proc. of IEEE INFOCOM*, Toronto, Canada, pp. 1851–1859, 2014.
- [20] J. Y. Shin, B. Wong and E. G. Sirer, "Small-world datacenters," in *Proc. of ACM SOCC*, Cascais, Portugal, pp. 1–13, 2011.
- [21] D. Li, H. Qi, Y. Shen and K. Li, "DPCell: Constructing novel architectures of data center networks on dual-port servers," *IEEE Network*, vol. 35, no. 4, pp. 206–212, 2021.
- [22] Z. Zhang, Y. Deng, G. Min, J. Xie, L. T. Yang *et al.*, "Hsdcc: A highly scalable data center network architecture for greater incremental scalability," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 5, pp. 1105–1119, 2019.
- [23] Das and K. Sajal, "Book review: Introduction to parallel algorithms and architectures: Arrays, trees, hypercubes by F. T. Leighton," *Acm Sigact News*, vol. 23, no. 3, pp. 31–32, 1992.
- [24] T. W. Kim, Y. Pan and J. H. Park, "Otp-based software-defined cloud architecture for secure dynamic routing," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 1035–1049, 2022.
- [25] T. Song, Z. Jiang, Y. Wei, X. Shi, X. Ma *et al.*, "Traffic aware energy efficient router: Architecture, prototype and algorithms," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3814–3827, 2016.